

Drawing Sheet 1 of 9
 Title: XML TYPES IN JAVA
 Inventor: David Bau
 Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
 Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
 Filing date: Filed Herewith Express Mail No.: EV 327619127 US

```
<xsd:schema . . .>
<xsd:element name="purchase-order">
<xsd:complexType>
<xsd:element name="line-item" maxOccurs="unbounded">
<xsd:complexType>
<xsd:element name="desc" type="xsd:string"/>
<xsd:element name="itemid" type="xsd:int"/>
<xsd:element name="price" type="xsd:float"/>
<xsd:element name="qty" type="xsd:int"/>
</xsd:complexType>
</xsd:element>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Figure 1

```
/**
/** @xbean:schema-type element="purchase-order" */
interface PurchaseOrder implements XmlObject
{
    /**
     * @xbean:schema-type element="line-item"
     */
    interface LineItem implements XmlObject
    {
        String getDesc();
        void setDesc(String desc);
        int getItemid();
        void setItemid(int itemid);
        float getPrice();
        void setPrice(float price);
        int getQty();
        void setQty(int qty);
    }
    LineItem getLineItem(int index);
    void setLineItem(int index, LineItem lineItem);

    int getLineItemCount();

    void addLineItem(LineItem lineItem);
    void insertLineItem(int index, LineItem lineItem);
    LineItem insertNewLineItem(int index);

    void removeLineItem(int index);
}
```

Figure 3

Title: XML TYPES IN JAVA
 Inventor: David Bau
 Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
 Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
 Filing date: Filed Herewith Express Mail No.: EV 327619127 US

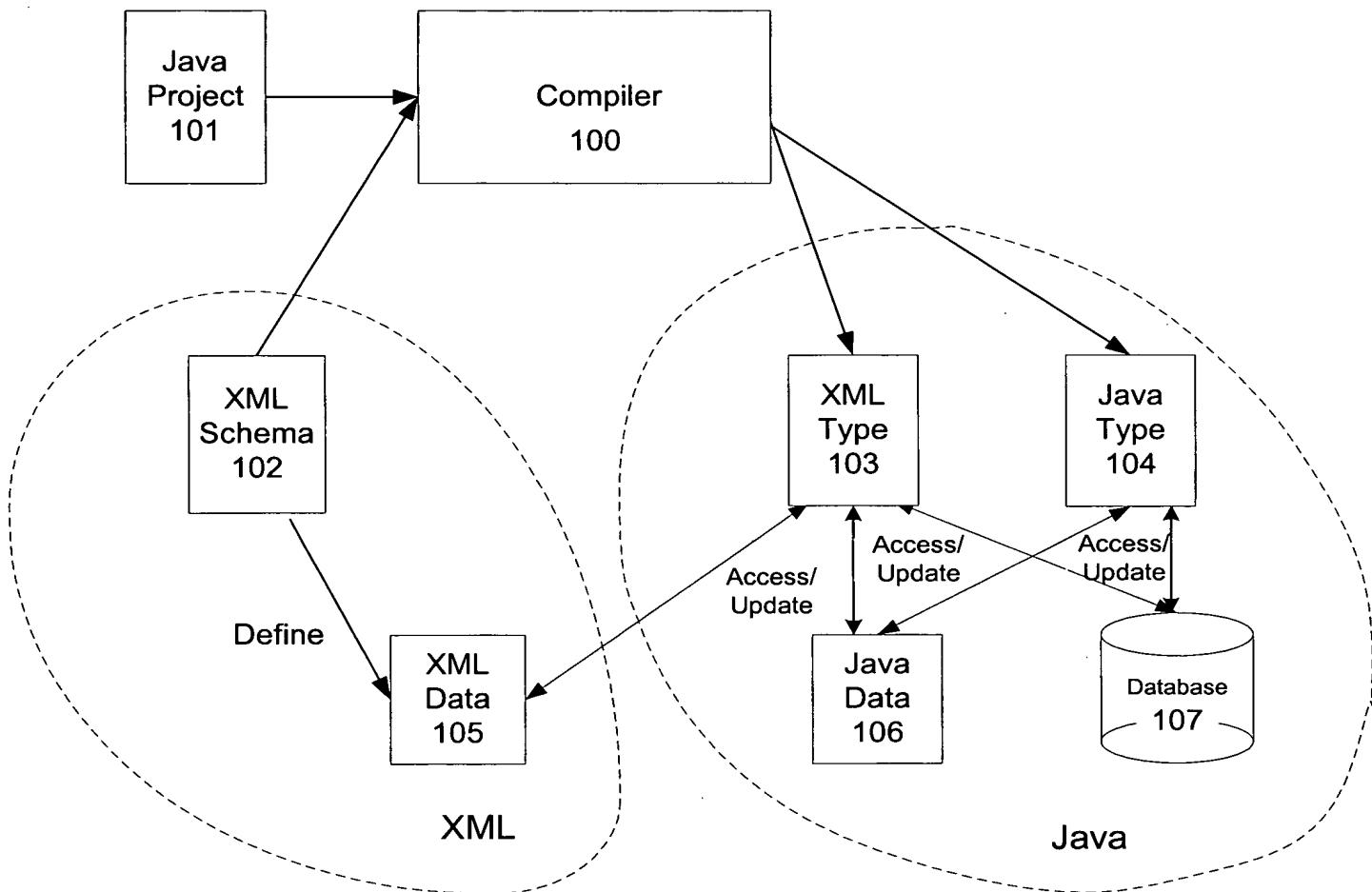


Figure 2

Drawing Sheet 3 of 9
 Title: XML TYPES IN JAVA
 Inventor: David Bau
 Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
 Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
 Filing date: Filed Herewith Express Mail No.: EV 327619127 US

```

        }

    /**
     * @xb:schema
     * element::
     *   <xsd:element name="myData">
     *     <xsd:sequence>
     *       <xsd:element name="a" type="xsd:string">
     *       <xsd:element name="i" type="xsd:int">
     *     </xsd:sequence>
     *   </xsd:element>
     *   :::
     * @xb:xml-map
     * map::
     *   <myData>
     *     <a>{a}</a>
     *     <i>{i}</i>
     *   </myData>
     *   :::
     */
    class MyData implements XBean
    {
        String getA();
        void setA(String a);

        int getI();
        void setI(int i);
    }
}

```

Figure 4

```

1  /** @jws:operation */
2 PurchaseOrder fixQuantities(PurchaseOrder po)
3 {
4     LineItem[] emptyItems;
5     items = (LineItem[])po.getAllValues("//line-item[qty=0]");
6     for (int i = 0; i < emptyItems.length; i++)
7         EmptyItems[i].setQty(1);
8     return po;
9 }

```

Figure 5

Drawing Sheet 4 of 9
Title: XML TYPES IN JAVA
Inventor: David Bau
Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
Filing date: Filed Herewith Express Mail No.: EV 327619127 US

```
1 /** @jws:operation */  
2 PurchaseOrder fixQuantities(PurchaseOrder po)  
3 {  
4     LineItem[] emptyItems;  
5     items = (LineItem[])po.getAllValues("//line-item[qty=0]");  
6     for (int i = 0; i < emptyItems.length; i++)  
7         emptyItems[i].set(1);  
8     items = (LineItem[])po.getAllValues("//line-item");  
9     for (int i = 0; i < emptyItems.length; i++)  
10    {  
11        int itemid = emptyItems[i].getItemid();  
12        emptyItems[i].set(findCatalogItem(emptyItems[i]));  
13    }  
14    return po;  
15 }
```

Figure 6

```
<xs:type name="CatalogItem">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element name="catalogID" type="xs:int"/>  
      <xs:element name="description" type="xs:string"/>  
      <xs:element name="price" type="xs:float"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:type>
```

Figure 7

Drawing Sheet 5 of 9
Title: XML TYPES IN JAVA
Inventor: David Bau
Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
Filing date: Filed Herewith Express Mail No.: EV 327619127 US

```
<xt:transformation
  xmlns:dbtypes="java://dbtypes.mycompany.com/"
  xmlns:po="http://standards.org/po-152"
  xmlns:xsi="... more namespaces ..."
<xt:input>
  <xt:argument name="input" type="dbtypes:CatalogID"/>
</xt:input>
<xt:output type="po:LineItem">
<xt:xquery><![CDATA[
  <x:value xsi:type="po:LineItem">
    <pi:description>{$input/description}</pi:item-id>
    <pi:itemid>{$input/catalogID}</pi:item-id>
    <pi:price>{$input/price}</pi:item-id>
    <pi:qty>0</pi:qty>
  </x:value>
]]></xt:xquery>
</xt:output>
</xt:transformation>
```

Figure 8

```
9  for (int i = 0; i < emptyItems.length; i++)
10 {
11     int itemid = emptyItems[i].getItemid();
12     int rememberQty = emptyItems[i].getQty();
13     emptyItems[i].set(findCatalogItem(emptyItems[i]));
14     emptyItems[i].setQty(rememberQty);
15 }
```

Figure 9

Drawing Sheet 6 of 9
Title: XML TYPES IN JAVA
Inventor: David Bau
Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
Filing date: Filed Herewith Express Mail No.: EV 327619127 US

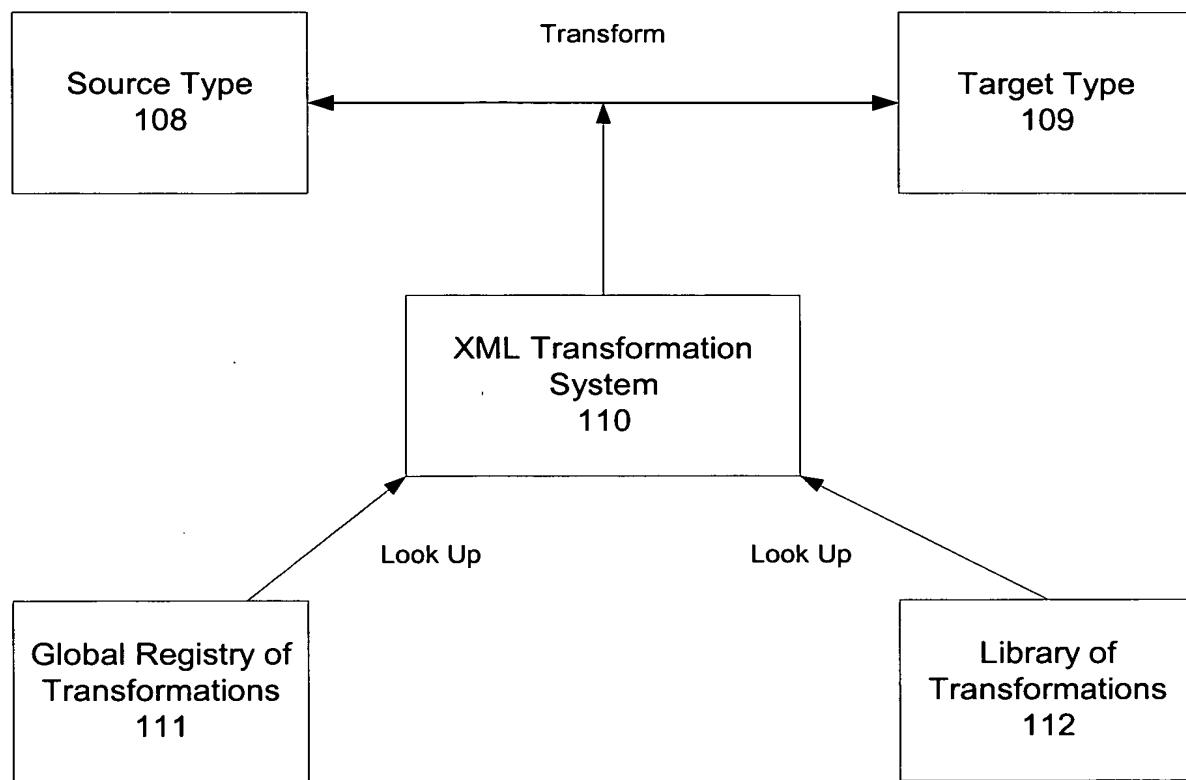


Figure 10

Drawing Sheet 7 of 9
XML TYPES IN JAVA

Title: David Bau
 Inventor: David Bau
 Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
 Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
 Filing date: Filed Herewith Express Mail No.: EV 327619127 US

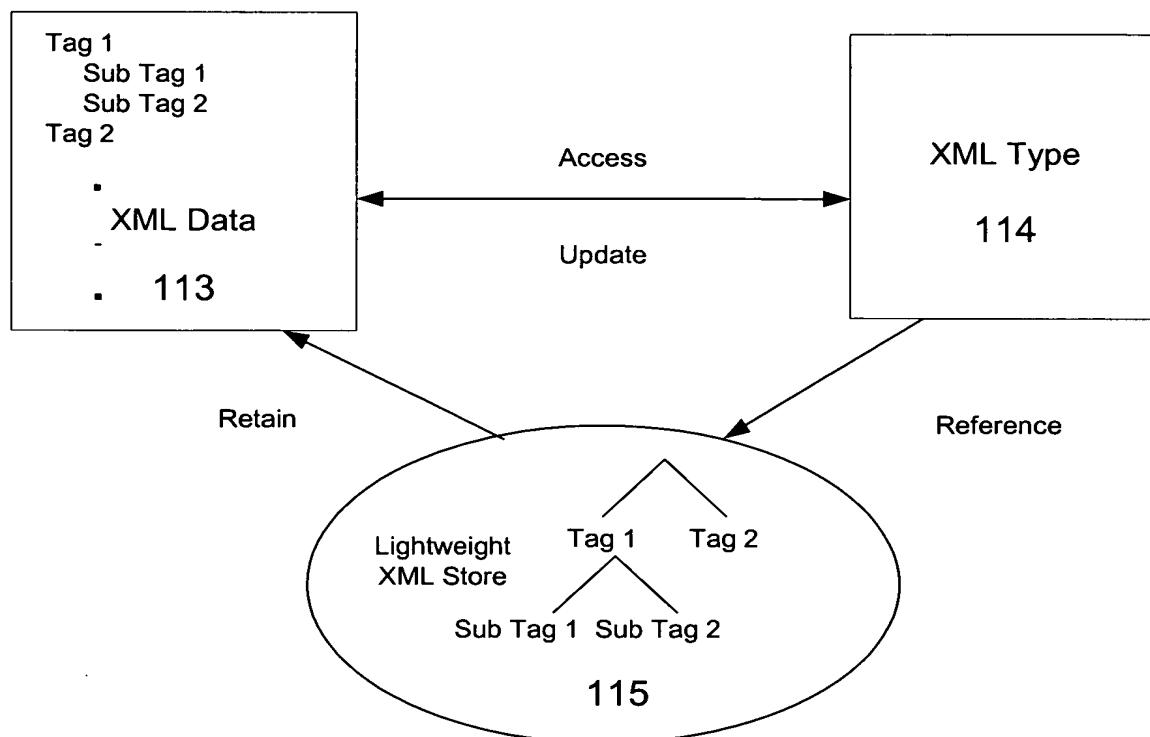


Figure 11

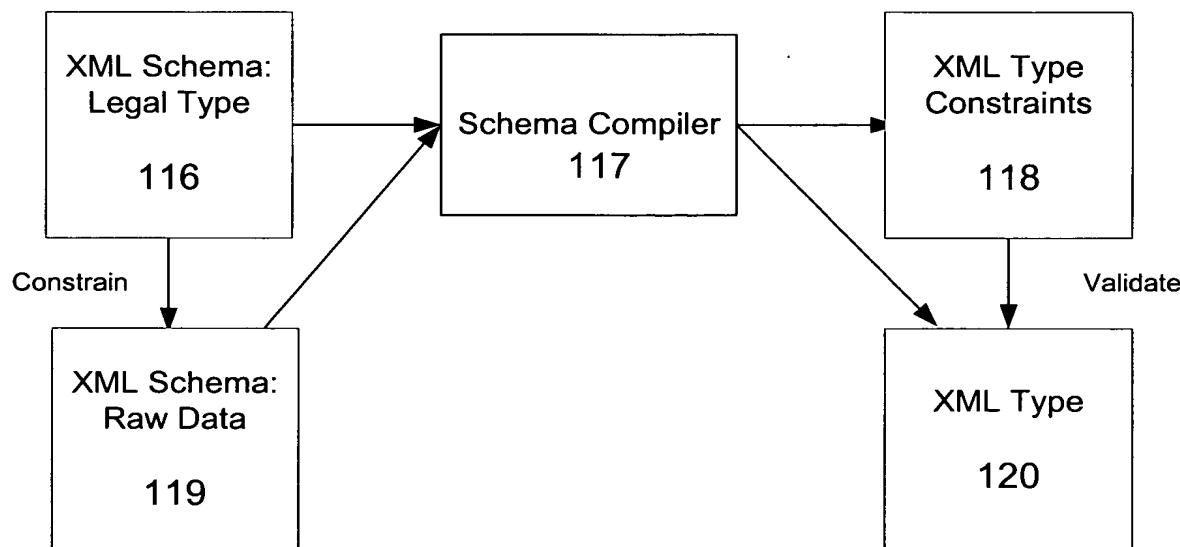


Figure 12

Drawing Sheet 8 of 9
Title: XML TYPES IN JAVA
Inventor: David Bau
Application No.: Not Yet Assigned Attorney: Sheldon R. Meyer
Docket No.: BEAS-01388US1 Telephone No.: (415) 362-3800
Filing date: Filed Herewith Express Mail No.: EV 327619127 US

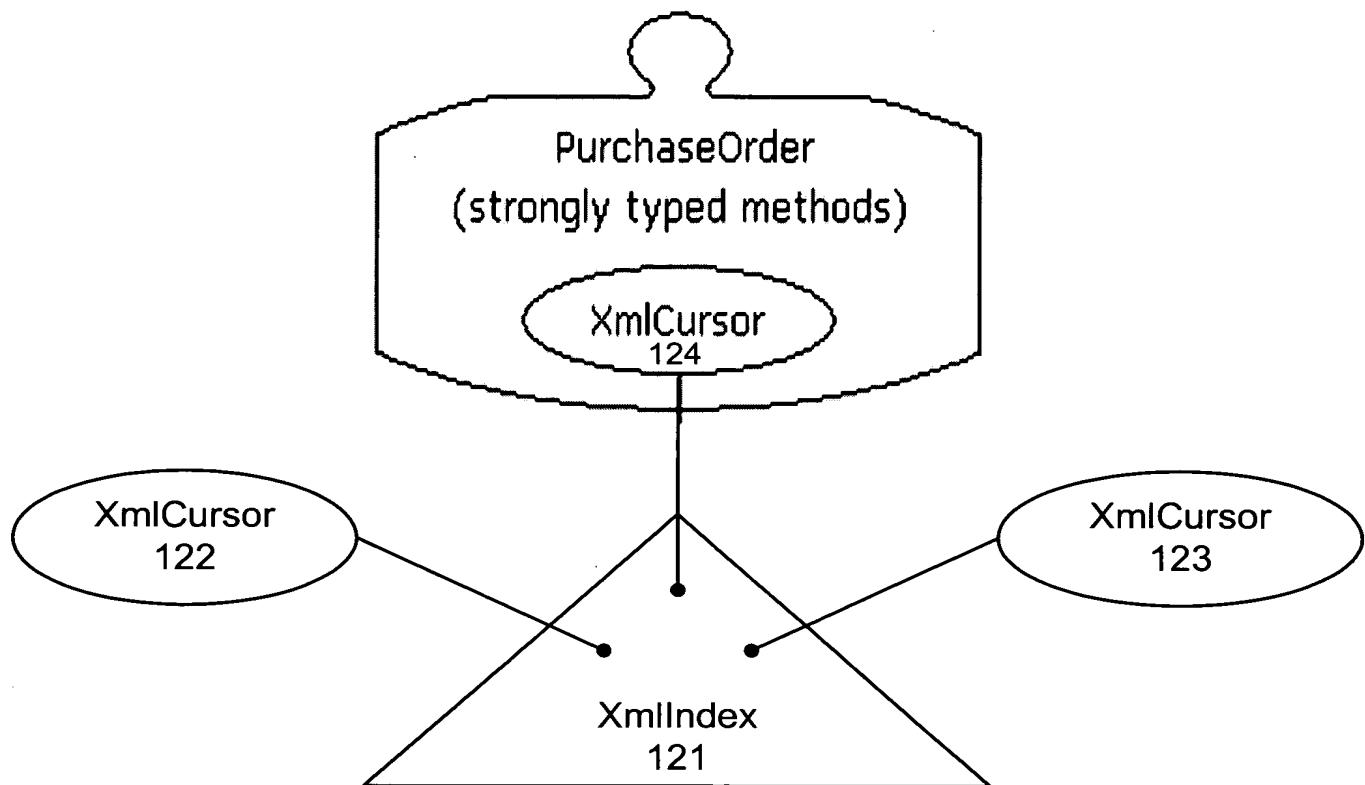


Figure 13

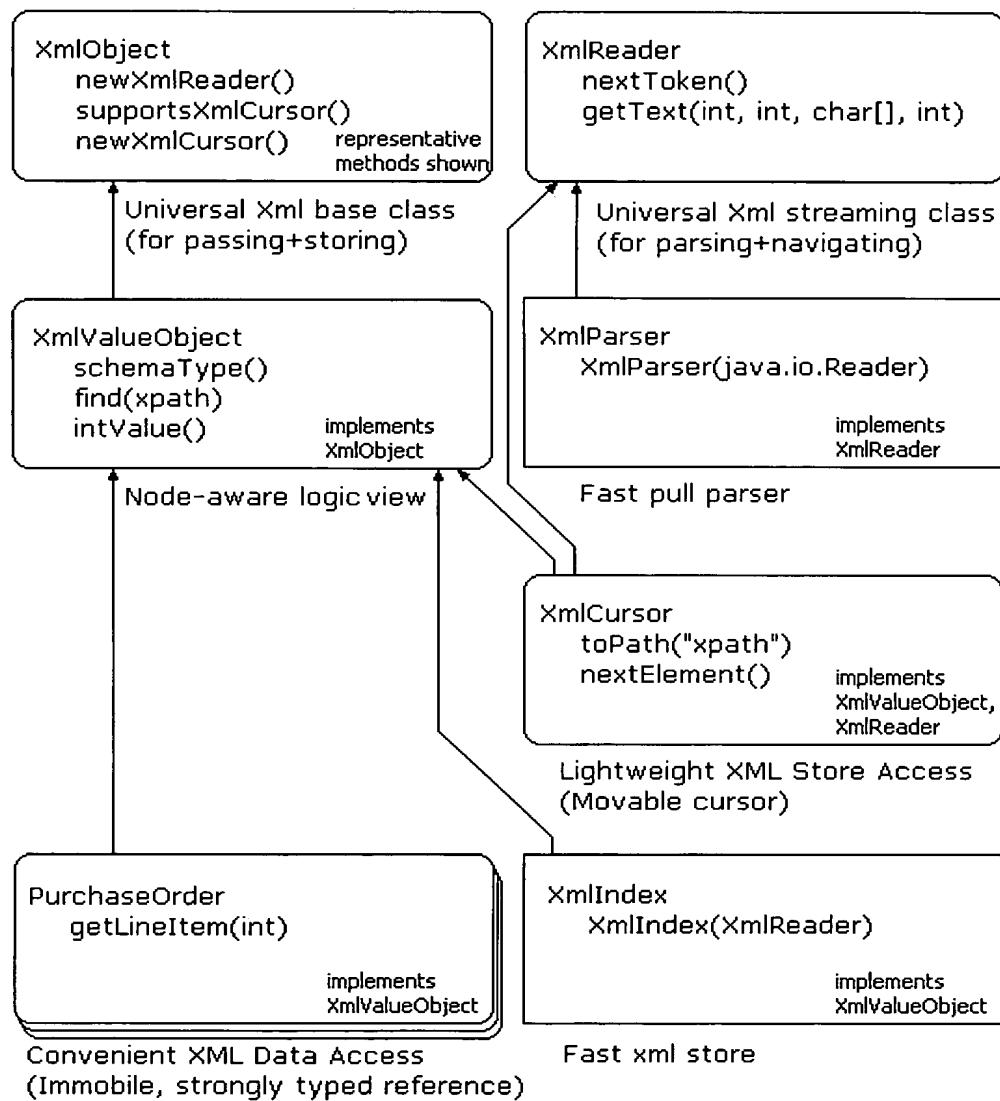


Figure 14